

# LINEAR REGRESSION WITH AMAZON AWS MACHINE LEARNING



Here we show how to use Amazon AWS Machine Learning to do linear regression. In a previous post, we explored using [Amazon AWS Machine Learning for Logistic Regression](#).

To review, **linear regression** is used to predict some value  $y$  given the values  $x_1, x_2, x_3, \dots, x_n$ . In other words it finds the coefficients  $b_1, b_2, b_3, \dots, b_n$  plus an offset  $c$  to yield this formula:

$$y = b_1x_1 + b_2x_2 + b_3x_3 + \dots + c.$$

It uses the **least squares error** approach to find this formula. In other words, think of all these values  $x_1, x_2, \dots$  existing in some  $N$ -dimensional space. The line  $y$  is the line that minimizes the distance between the observed and predicted values for all these values. So it is the line that most nearly split right down the middle of the data observed in the training set. Since we know what that line looks like, we can take any new data, plug those into the formula, and then make a prediction.

As always models are built like this:

- Take an input set of data that you think it correlated. Such as hours of exercise and weight reduction.
- Split that data into a training set and testing set. Amazon does that splitting for you.
- Run the linear regression algorithm to find the formula for  $y$ . Amazon picks linear regression based upon the characteristics of the data. It would pick another type of regression or classification model if we picked a data set that for which that was a better fit.

- Check how accurate the model is by taking the square root of the differences between the observed and predicted values. Amazon actually uses the mean of this difference.
- Then take new data and apply the formula  $y$  to make a prediction.

## Get Some Data

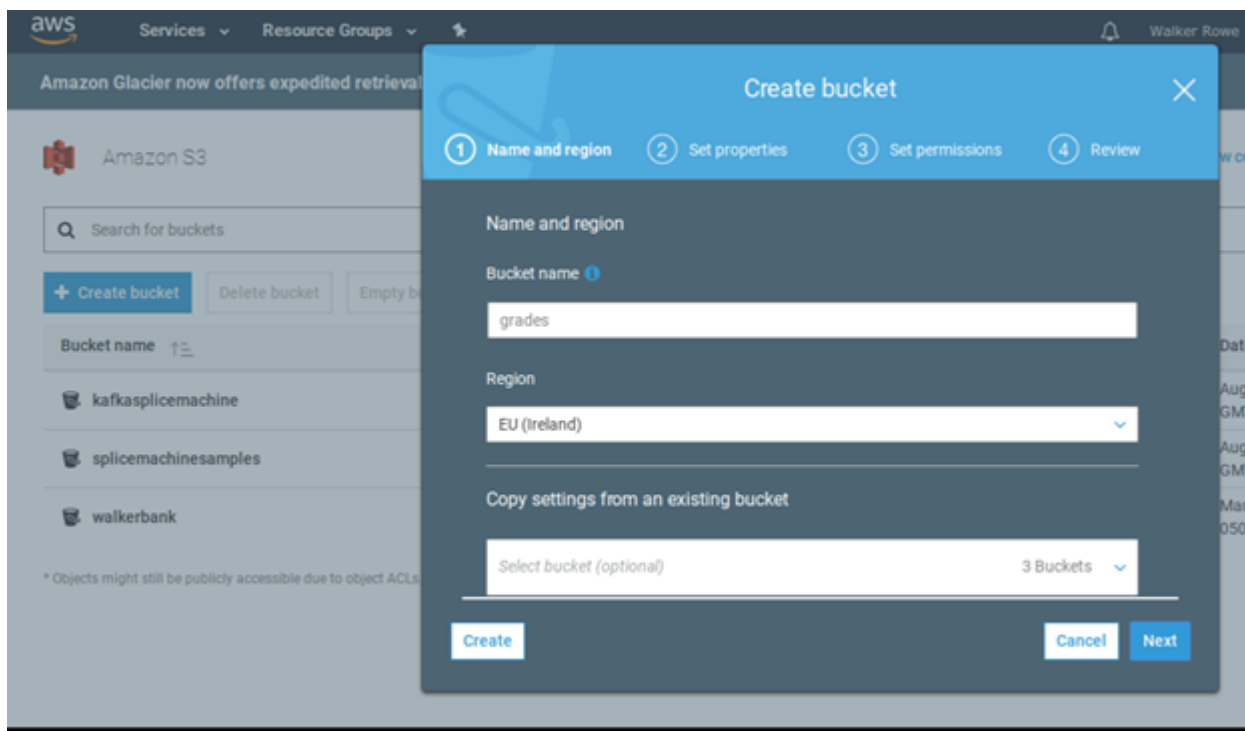
We will use [this data](#) of student test scores from the UCI Machine Learning repository.

I copied this data into Google Sheets [here](#) so that you can more easily read it. Plus I show the training data set and the one used for prediction.

You download [this data in raw format](#) and upload it to Amazon S3. But first, we have to delete the column headings and change the semicolon (;) separators to commas (,) as shown below. We take the first 400 rows as our model training data and the last 249 for prediction. Use `vi` to delete the first from the data as Amazon will not read the schema automatically (Too bad it does not).

```
vi student-por.csv
sed -i 's/;/,/g' student-por.csv
head -400 student-por.csv > grades400.csv
tail -249 student-por.csv > grades249.csv
```

Now create a bucket in S3. I called it gradesml. Call yours some different name as it appears bucket names have to be unique across all of S3.



Then upload all 3 files.

Amazon S3 > gradesml

Overview

Properties

Per

 Upload

 Create folder

More 

This bucket is empty

and make sure the permissions are set to **read**. Note the https link

# student-por.csv Latest version ▾

Overview

Properties

Permissions

Open

Download

Download as

Make public

Copy path

**Owner**

critique\_american

**Last modified**

Mar 15, 2018 11:35:12 AM GMT-0400

**Etag**

973e7bacf3a16bd16e92185dfd64706b

**Storage class**

Standard

**Server side encryption**

None

**Size**

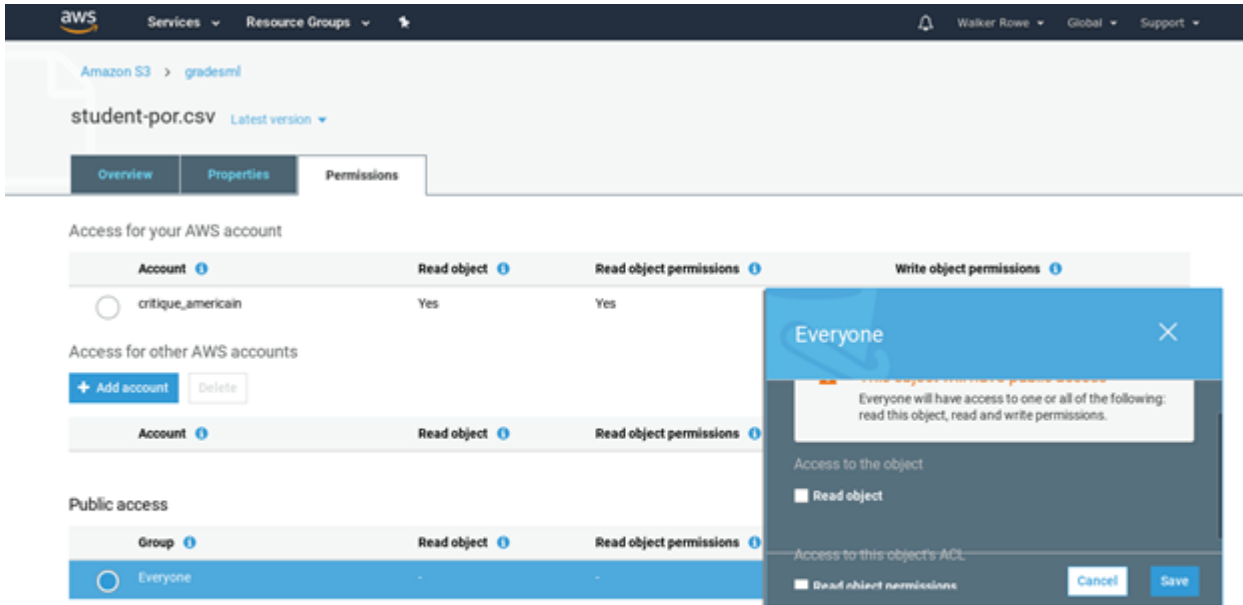
93220

**Link**

<https://s3-eu-west-1.amazonaws.com/gradesml/student-por.csv>


Give read



permissions:





Click on **Amazon Machine Learning** and then **Create New Data Source/ML Model**. If you have not used ML before it will ask you to sign up. Creating and evaluating models is free. Amazon charges you for using them to make prediction on a per 1,000 record basis.

orange). Group A

-  **Machine Learning**
- Amazon SageMaker
- Amazon Comprehend
- AWS DeepLens
- Amazon Lex
- Machine Learning
- Amazon Polly
- Rekognition
- Amazon Transcribe
- Amazon Translate

-  **AR & VR**
- Amazon Sumerian [↗](#)
-  **Application Integration**
- Step Functions
- Amazon MQ
- Simple Notification Service
- Simple Queue Service
- SWF

-  **Analytics**
- Athena
- EMR
- CloudSearch

-  **Customer Engagement**
- Amazon Connect
- Pinpoint
- Simple Email Service

Click create new

**Datasource and ML model.**

## Objects

Create new... Actions

- Datasource and ML model
- Datasource
- ML model
- Evaluation
- Batch prediction

	Name or ID	Type	ID
<input type="checkbox"/>	ML model: banking	Batch prediction	bp-z56xF
<input type="checkbox"/>	Evaluation: ML model: banking	Evaluation	ev-o5yt6:
<input type="checkbox"/>	ML model: banking	ML model	ml-BwZ5
<input type="checkbox"/>	banking (percentBegin=70, percentEnd=10...	Datasource	ds-TWFF

Fill in the **S3 location** below. Notice that you do not use the URL. Instead, put the bucket name and file name:

Click **verify** and **Grant Permissions** on the screen that pops up next.

How to access your data and give it permission to access it.

**S3 location \***

Enter the path to a single file or folder in Amazon S3. You need to grant Amazon ML permission to read this data. [Learn more.](#)

If you already have a schema for this data, provide it in a file at s3://<path-of-input-data>.schema. If you don't have a schema, Amazon ML will help you create one on the next page. ⓘ

**source name**

The validation is successful. To go to the next step, choose Continue

**Datasource name** Student-por.csv

**Data location** s3://gradesml/grades.csv

**Data format** CSV

**Schema source** Auto generated

**Number of files** 1

**Total size** 67.6 KB

\* Required

Give the data source some name then click through the screens. It will also determine whether each value is categorical (drawn from a finite set) or just a number. What is important for you to do is to pick the **target**. That is the dependant value you want it to predict, i.e., **y**. From the input data **student-por.csv** pick **G3**, as that is the student's final grade. These grades are from the Portuguese grammar school system and 13 is the highest value.

Below **don't use** students-por.csv as the input data. Instead use **grades400.csv**.

1. **Input Data** 2. Schema 3. Target 4. Row ID 5. Review

## Input data

Import your data to create an Amazon ML datasource. Amazon ML can use your datasource to create and evaluate an ML model, and you can use

Where is your data?



Amazon Redshift

### S3 data access

Tell Amazon ML how to access your data and give it permission to access it.

S3 location \*

s3:// gradesml/student-por.csv

Enter the path to a single file or folder in Amazon S3. You need to grant Amazon ML permission to read this data. [Learn more.](#)

If you already have a schema for this data, provide it in a file at s3://<path-of-input-data>-schema. If you don't have a schema, Amazon ML will help you create one on the next page.

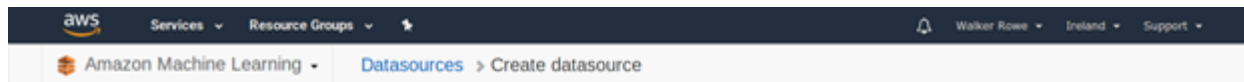
Datasource name

\* Required

Reset

Cancel

Verify



1. Input Data 2. **Schema** 3. Target 4. Row ID 5. Review

## Schema

Amazon ML scanned your input data and inferred the column names and data type for each of the columns in your dataset. Review and edit the data type for each column to ensure that it accurately represents the data. This enables Amazon ML to read the input data correctly and to produce accurate predictions. [Learn more.](#)

Does the first line in your CSV contain the column names?  Yes  No

ACTION: Change type

	Name	Data type	Sample field value 1	Sample field value 2	Sample field value 3
1	Var01	Categorical	GP	GP	GP
2	Var02	Categorical	F	F	F
3	Var03	Numeric	18	17	15
4	Var04	Categorical	U	U	U
5	Var05	Categorical	GT3	GT3	LE3
6	Var06	Categorical	A	T	T
7	Var07	Numeric	4	1	1
8	Var08	Numeric	4	1	1
9	Var09	Categorical	at_home	at_home	at_home
10	Var10	Categorical	teacher	other	other

Cancel




Previous

Continue

Now Amazon

builds the model. This will take a few minutes.

## ML model summary

<b>ID</b>	ml-Q5G6ld7g7Xj
<b>Name</b>	ML model: training 
<b>Type</b>	Numerical regression
<b>Creation time</b>	Mar 15, 2018 12:17:24 PM
<b>Completion time</b>	Not available 
<b>Compute Time (Approximate)</b>	Not available 
<b>Status</b>	Pending
<b>Log</b>	Not available

## Datasource (training)

<b>Datasource ID</b>	ds-E9YJUuZONWU
<b>Target</b>	_Target_
<b>Input schema</b>	<a href="#">View input schema</a>

## Evaluations

<b>Evaluations created</b>	1
<b>Latest evaluation result</b>	Not available

[Perform another Evaluation](#)

## Predictions

CloudWatch metrics [View in CloudWatch](#)

### A single dataset

Generate one-time predictions for a single dataset.

[Generate batch predictions](#)

### Try real-time predictions

Generate real-time predictions in your browser.

[Try real-time predictions](#)

### Enable real-time predictions

To enable real-time predictions now, create a real-time prediction endpoint.

[Create endpoint](#)

While waiting are create another **data set**. This is not a model so it will not ask you for a target. Use the **grades249.csv** file in S3, which we will use in the **batch prediction** step.



## Objects

Create new... Actions

Filter: All types  Items per page

	Name	Type	ID	Status
<input type="checkbox"/>	prediction	Datasource	ds-yBotR7rXRo5	In progress
<input type="checkbox"/>	Evaluation: ML model: training	Evaluation	ev-1XUCxHi1MzG	Completed
<input type="checkbox"/>	ML model: training	ML model	ml-Q5G6ld7g7Xj	Completed
<input type="checkbox"/>	training: percentBegin=70 percentEnd=1	Datasource	ds-8yI0c76RB8r	Completed

Now the evaluation is done. We can see which one it is from the list above as it says **evaluation**. Click on it. We explain what it means below.

## Evaluation Summary

ID	ev-1XUCxHi1MzG
Name	Evaluation: ML model: training
Datasource ID	ds-8yI0c76RB8r
Output location	Not available
Creation time	Mar 15, 2018 12:17:24 PM
Completion time	3 mins.
Compute Time (Approximate)	2 mins.
Status	Completed
Log	<a href="#">Download log</a>

## ML model performance

On your most recent evaluation, **ev-1XUCxHi1MzG**, the ML model's quality score is **better** than the baseline.

**RMSE: 1.7457**  
 RMSE baseline: 2.933  
 Difference: 1.187



[Explore model performance](#)

Tags [Add or edit tags](#)

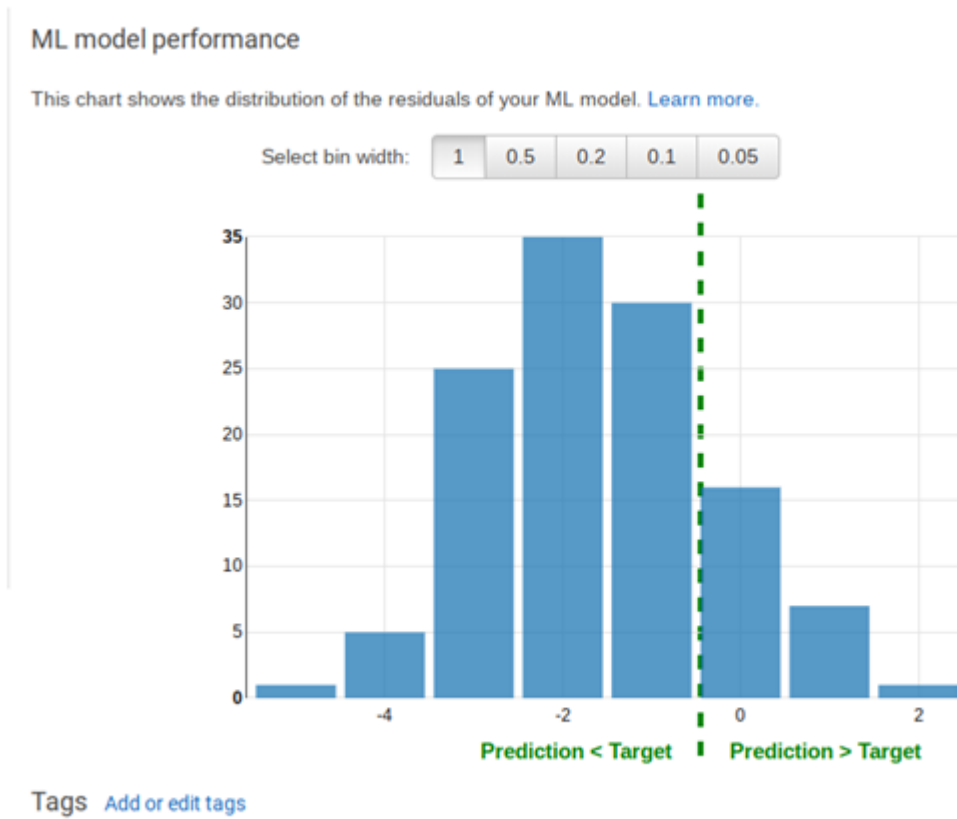
Amazon shows the RMSE. This is the square root of the sum of the squared differences of the observed and predicted values. We square and then take the square root so that all the numbers are positive, so they do not cancel each other out. Amazon also uses the mean, meaning average, by multiplying this sum by  $1/n$ , where  $n$  is the sample size.

If the model and the evaluations were the same, this number would be 0. So the closer to 0 zero we get the more accurate is our model. If the number is large, then the problem is not the algorithm, it is the data. So we could not pick another algorithm to make it much better. There is really only one algorithm used for LR, finding the least squares error. (There are more esoteric ones.) If MSE number is large then either the data is not correlated or, more like, most of the data is correlated, but some

of it is not and is thus messing up our model. What we would do is drop some columns out and rebuild out model to get a more accurate model.

What value means the model is good? The model is good when the distribution of errors is a normal distribution, i.e., the bell curve.

Put another way, click **Explore Model Performance**.



See the histogram above. Numbers to the left of the dotted line are where the predicted values were less than the observed ones. Numbers to the right are where they are higher. If this distribution were entered on the number 0 then we would have a completely random distribution. That is the idea situation where our errors are distributed randomly. But since it is shifted there is something in our data that we should leave out. For example, family size might not be correlated to grades.




Above Amazon showed the **RMSE baseline**. This is what the RMSE would be if we could have an input data set in which there was this perfect distribution of errors.

Also here we see the limitations of doing this kind of analysis in the cloud. If we have written our own program we could have calculated other statistics that showed exactly which column was messing up our model. Also we could try different algorithms to get rid of the bias caused by **outliers**, meaning numbers far from the mean that distort the final results.

## Run the Prediction

Now that the model is saved, we can use it to make predictions. In other words we want to say given these student characteristics what are their likely final grades going to be.

Select the **prediction** datasource you created above then select **Generate Batch Predictions**. Then click through the following screens.

**ID** ds-yBotR7rXRo5  
**Name** prediction   
**Creation time** Mar 15, 2018 12:25:39 PM  
**Completion time** 4 mins.   
**Compute Time (Approximate)** 13 mins.   
**Status** **Completed**  
**Message** Not available  
**Input schema** [View input schema](#)  
**Log** [Download log](#)

Use this datasource to ▾

Copy settings to create a new datasource  
Create (train) an ML model  
Evaluate an ML model  
Generate batch predictions

**S3 location**  
**Number of files**  
**Data format** CSV  
**Total size** 34.6 KB  
**Data rearrangement** None

**1. ML model for batch prediction** 2. Data for batch prediction 3. Batch prediction results 4. Review

## ML model for batch prediction

Choose the ML model to use for generating batch predictions. Batch predictions generate predictions all at once for a large number of data records


Select an ML model

Search All ML models by name or ID

**ML model name** ML model: training Change ML model

<b>ML model ID</b> ml-Q5G6ld7g7Xj	<b>Input schema</b> <a href="#">View input schema</a>
<b>ML model type</b> Numerical regression	<b>Target attribute</b> _Target_
<b>Creation time</b> Mar 15, 2018 12:17:24 PM	<b>Target type</b> NUMERIC
<b>Status</b> Completed	<b>Number of attributes</b> 33
<b>Datasource ID</b> ds-E9YJUuZ0NWU	<b>Evaluations created</b> 1
<b>Log</b> <a href="#">Download log</a>	<b>Latest evaluation result</b> 1.746 (RMSE)
	<b>Batch predictions created</b> 0

Tags  
No tags

 You selected ML model ml-Q5G6ld7g7Xj. To go to the next step, choose Continue

Cancel

Continue

# ML model settings

You can use the automatically suggested ML model settings, or you can choose to customize.

**ML model type** REGRESSION ⓘ

**ML model target**

**ML model name (Optional)**

**Select training and evaluation settings**

Recipes and training parameters control the ML model training process. You can select these settings for your ML model or use the defaults provided by Amazon ML. In either case, you can choose to have Amazon ML reserve a portion of the input data for evaluation. [Learn more.](#)

**Default (Recommended)**

- Generate a default recipe
- Use default training parameters
- Set aside 30% of your training data to evaluate the training
- Split the evaluation data sequentially ⓘ

**Custom**

- Modify the recipe Amazon ML generates
- Modify training parameters
- Randomly or sequentially split your evaluation data ⓘ

**Evaluation Name**

[Cancel](#) [Previous](#) [Review](#)

Click **review**

then **create ML model.**

1. ML model for batch prediction   **2. Data for batch prediction**   3. Batch prediction results   4. Review

## Data for batch prediction

Locate the input data to use for the batch prediction. [Learn more about S3 permissions.](#)

**Locate the input data**    I already created a datasource pointing to my S3 data    My data is in S3, and I need to create a datasource

You selected ML model ml-Q5G6ld7g7Xj

Q Enter the datasource name or ID

Datasource name prediction

Cha

<b>Datasource ID</b>	ds-yBotR7rXRo5	<b>Input schema</b>	<a href="#">View input schema</a>
<b>Creation time</b>	Mar 15, 2018 12:25:39 PM	<b>Target attribute</b>	
<b>Status</b>	Completed	<b>Target type</b>	
<b>Datasource type</b>	S3	<b>Number of attributes</b>	33
<b>S3 location</b>	s3://gradesml/grades249.csv	<b>Models trained</b>	0
<b>Data format</b>	CSV	<b>Evaluations created</b>	0
<b>Data rearrangement</b>	None	<b>Batch predictions created</b>	0

Tags

No tags

Here we tell it

where to save the results in S3. There it will save several files. The one we are interested in is the one where it calculates the **score**. It should tack it onto the input data to make it easier to read. But it does not. So I have pasted it into [this spreadsheet](#) for you on the sheet called prediction and added back the column headings. I also then added a column to show how the MSE mean squared error is calculated.

## Batch prediction results

The estimated cost for generating your predictions is **\$0.10**. This estimate is based on the 249 data records included in your prediction request.  
The Amazon ML fee for batch predictions is **\$0.10 per 1,000 predictions**, rounded up to the next 1,000. [Learn more.](#)

Type the path to the S3 location in which the prediction results will be saved.

S3 destination

s3:// gradesml/predictions.csv

Batch prediction name  
(Optional)

Batch prediction: ML model: training

Cancel

Previous

Review

Amazon Machine Learning > Batch Predictions > Create batch prediction

1. ML model for batch prediction 2. Data for batch prediction 3. Batch prediction results **4. Review**

### Review

Review and make any changes, and then click Finish.

**ML model for batch prediction** Edit

ML model Name	ML model: training
ML model ID	ml-Q5G6ld7g7Xj

**Data for batch prediction** Edit

Datasource name	prediction
Data location	s3://gradesml/grades249.csv

**Batch prediction results** Edit

Output location	s3://gradesml/predictions.csv
Batch prediction name	Batch prediction: ML model: training

**Cost Estimate**

The estimated cost for generating your predictions is **\$0.10**. This estimate is based on the 249 data records included in your prediction request.  
The Amazon ML fee for batch predictions is **\$0.10 per 1,000 predictions**, rounded up to the next 1,000. [Learn more.](#)

**Tags** 0




Amazon ML copies a maximum of 10 tags from parent objects. Edit the list to keep the tags you need.

No tags

Cancel Previous **Create batch prediction**

Feedback English (US) © 2008 - 2018, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

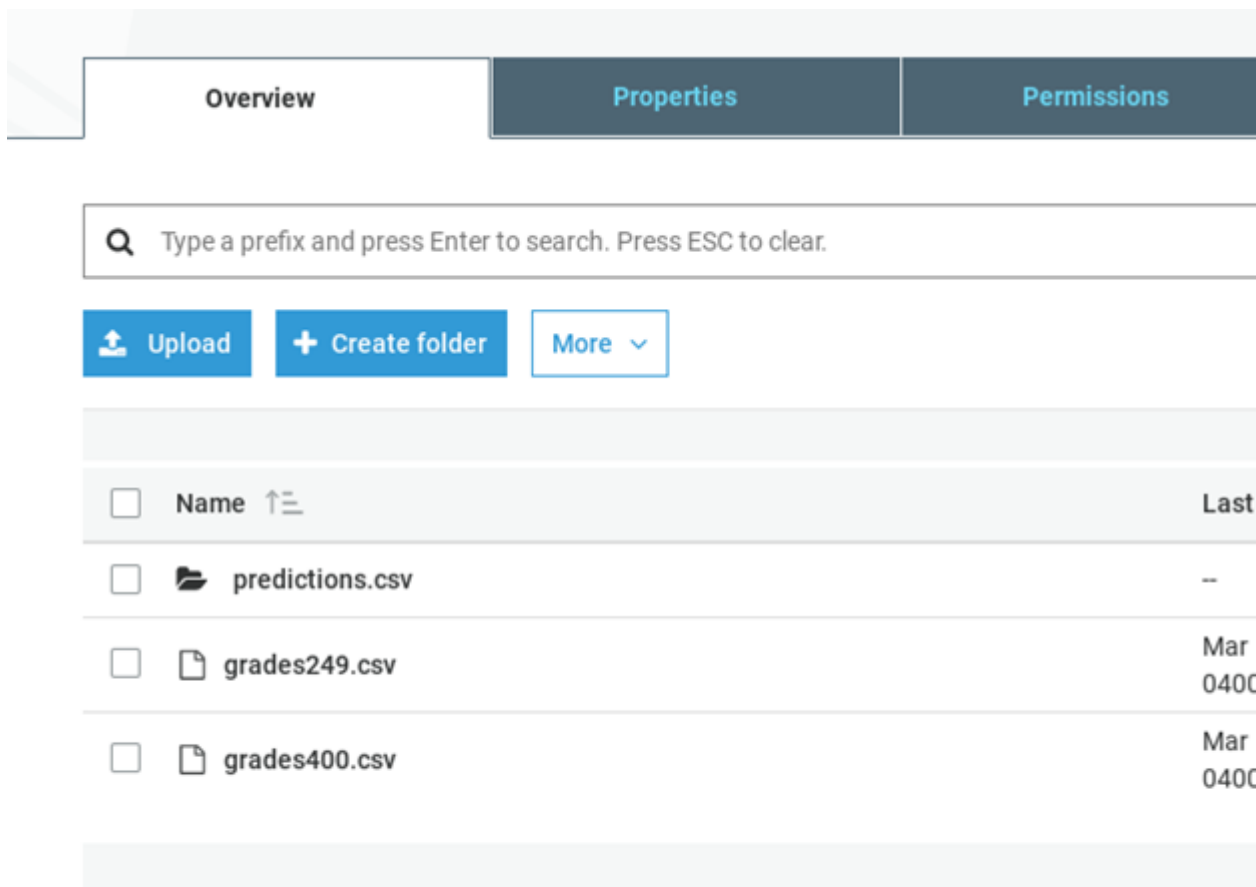
## Batch prediction summary

ID	bp-ebhjggKYchO
Name	Batch prediction: ML model: training 
Creation time	Mar 15, 2018 12:43:38 PM
Completion time	Not available 
Compute Time (Approximate)	Not available 
Status	In progress
Datasource ID	ds-yBotR7rXRo5
ML model ID	ml-Q5G6ld7g7Xj
Input S3 URL	s3://gradesml/grades249.csv
Output S3 URL	s3://gradesml/predictions.csv/
Log	Not available





## Processing information

Number of records seen	Not available
Records that failed to process	Not available

As you can see, it saves the data in S3 in a folder called **predictions.csv**. In this case it gave the prediction values in a file with this long name **bp-ebhjggKYchO-grades249.csv.gz**. You cannot view that online in S3. So download it showing the URL below and look at it with another tool. In this case I pasted the data into Google Sheets.



The screenshot shows the AWS S3 console interface. At the top, there are three tabs: 'Overview', 'Properties' (which is selected), and 'Permissions'. Below the tabs is a search bar with the text 'Type a prefix and press Enter to search. Press ESC to clear.' Underneath the search bar are three buttons: 'Upload', 'Create folder', and 'More'. The main area displays a list of files in a table format. The table has columns for 'Name' and 'Last modified'. The files listed are 'predictions.csv', 'grades249.csv', and 'grades400.csv'. The 'predictions.csv' file is highlighted in blue.

<input type="checkbox"/>	Name 	Last
<input type="checkbox"/>	 predictions.csv	--
<input type="checkbox"/>	 grades249.csv	Mar 15, 2018 12:43:38 PM
<input type="checkbox"/>	 grades400.csv	Mar 15, 2018 12:43:38 PM

bp-ebhjggKYch0-grades249.csv.gz Latest version ▾

Overview

Properties

Permissions

Open

Download

Download as

Make public

Copy path

**Owner**

amazon-machine-learning-admin+dub

**Last modified**

Mar 15, 2018 12:44:38 PM GMT-0400

**Etag**

6e04d3aa1bb4d0f2af9131e08dedb45e

**Storage class**

Standard

**Server side encryption**

None

**Size**

1155

**Link**

<https://s3-eu-west-1.amazonaws.com/gradesml/predictions.csv/batch-prediction/result/bp-ebhjggKYch0-grades249.csv.gz>

Download the

data like this:

wget

<https://s3-eu-west-1.amazonaws.com/gradesml/predictions.csv/batch-prediction/result/bp-ebhjggKYch0-grades249.csv.gz>

Here is that the data looks like with the prediction added to the right to make it easy to see. Column AG is the student's actual grade. AH is the predicted value. AI is the square of the difference. And then at the bottom is MSE.

AC	AD	AE	AF	AG	AH	AI
health	absences	G1	G2	G3	score	(obs-pred) sqrd
4	4	15	14	17	13.30	13.70998729
5	0	14	13	14	12.18	3.311817626
4	0	11	12	13	12.68	0.1035101929
1	10	12	15	15	14.13	0.7519317796
4	4	12	16	16	13.11	8.379114409
5	16	10	11	11	8.30	7.28465498
3	6	10	13	13	11.27	2.991585344
5	0	9	12	12	10.49	2.270657197
3	11	9	11	12	9.44	6.53817229
2	9	13	14	15	12.14	8.19110124
4	0	13	17	17	14.61	5.729368832
4	2	12	15	15	13.06	3.750729422
3	0	14	17	17	15.52	2.18750016
4	21	0	10	10	7.26	7.500108345

raw predict ▾

